FIG.1

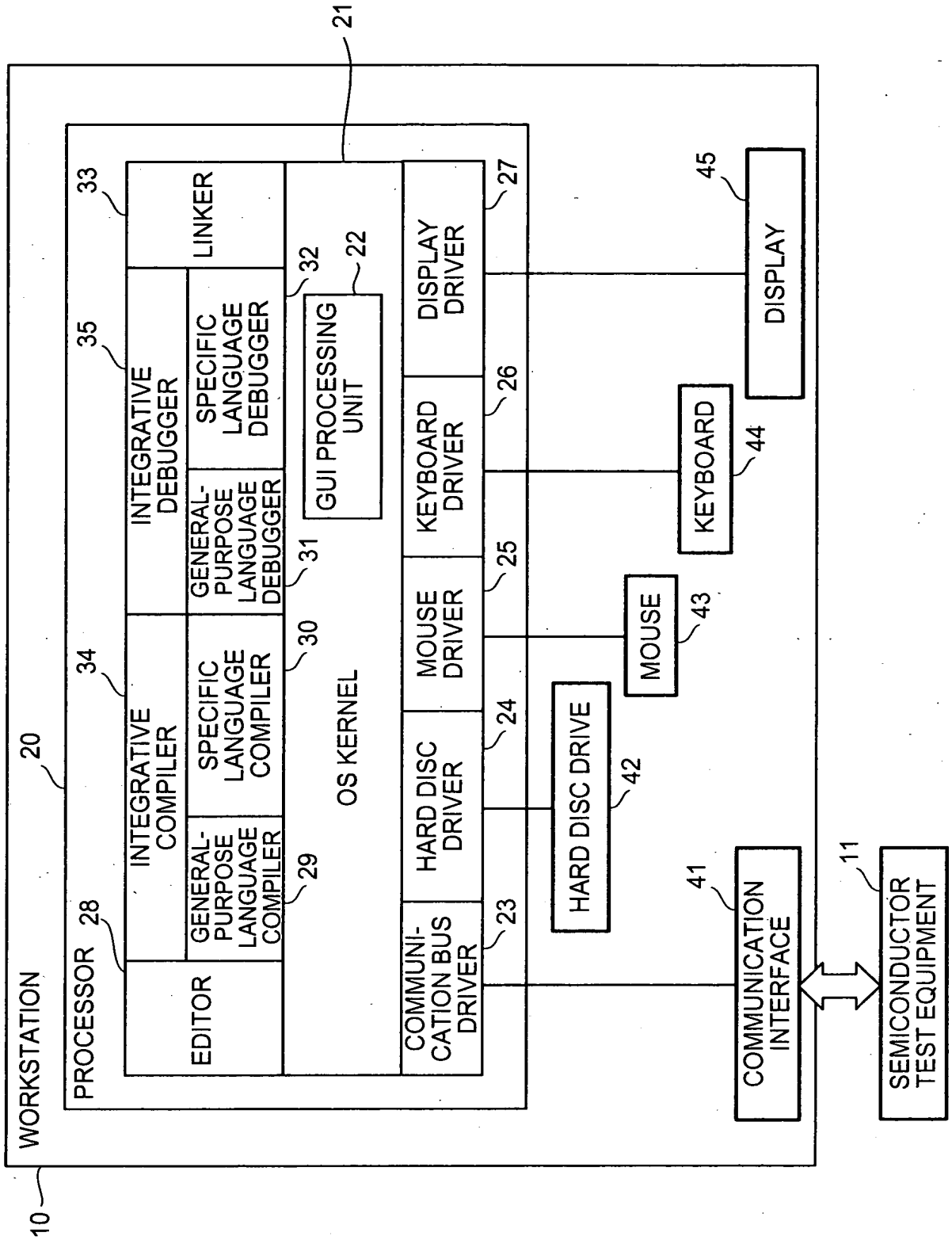

WORKSTATION
10
20
PROCESSOR
28
EDITOR
INTEGRATIVE COMPILER
34
GENERAL-PURPOSE LANGUAGE COMPILER
29
SPECIFIC LANGUAGE COMPILER
30
INTEGRATIVE DEBUGGER
35
GENERAL-PURPOSE LANGUAGE DEBUGGER
31
SPECIFIC LANGUAGE DEBUGGER
32
LINKER
33
OS KERNEL
21
GUI PROCESSING UNIT
22
COMMUNI-CATION BUS DRIVER
23
HARD DISC DRIVER
24
MOUSE DRIVER
25
KEYBOARD DRIVER
26
DISPLAY DRIVER
27
HARD DISC DRIVE
42
MOUSE
43
KEYBOARD
44
DISPLAY
45
COMMUNICATION INTERFACE
41
SEMICONDUCTOR TEST EQUIPMENT
11

FIG.2

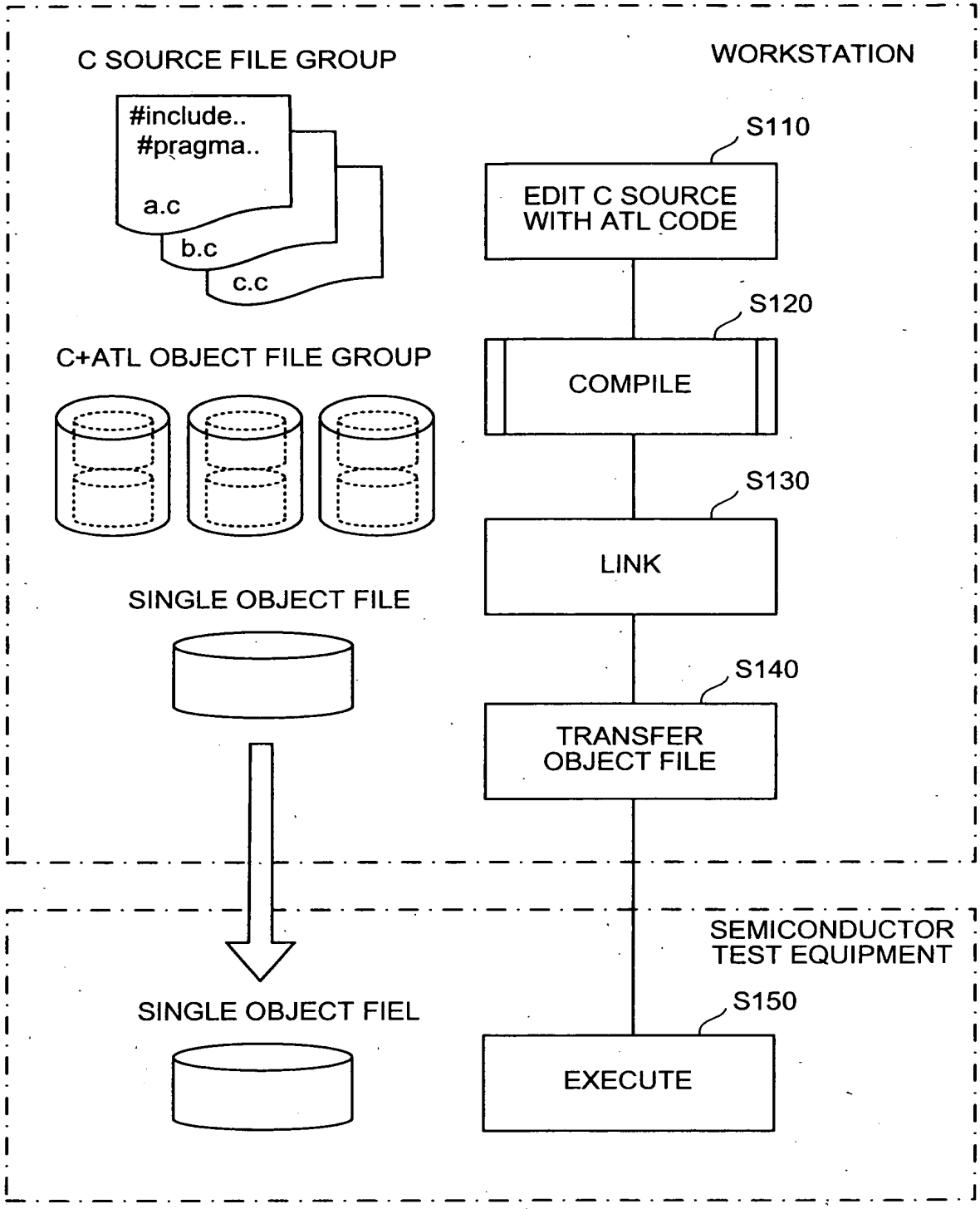
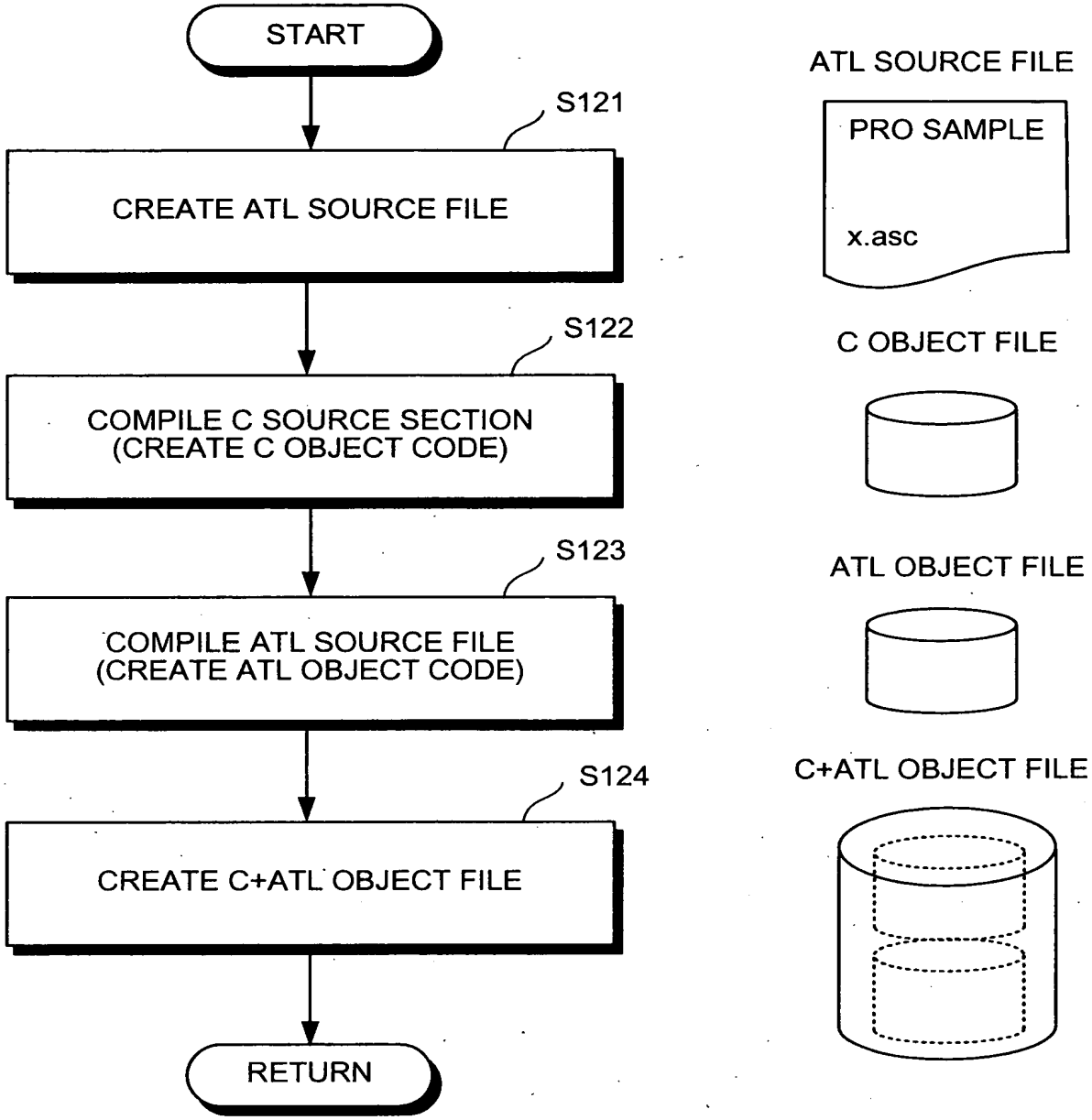

C SOURCE FILE GROUP
#include..
#pragma..
a.c
b.c
c.c
C+ATL OBJECT FILE GROUP
SINGLE OBJECT FILE
SINGLE OBJECT FIEL
WORKSTATION
S110
EDIT C SOURCE WITH ATL CODE
S120
COMPILE
S130
LINK
S140
TRANSFER OBJECT FILE
SEMICONDUCTOR TEST EQUIPMENT
S150
EXECUTE

FIG.3

```
1:      #include "AT/hybrid.h"
2:
3:      #pragma atl name      "SAMPLE"
4:      #pragma atl socket    "SSOC"
5:      #pragma atl proc P1(ARG1,ARG2(2)) "{
6:           WRITE "ARG1=",ARG1,/
7:           WRITE "ARG2=",ARG2,/
8:      }"
9:
10:     int test_main(void)
11:     {
12:      int idata;
13:      struct atreal rdata[2];
14:
15:      if(ATL_init2(EM_ATL) != 0){
16:         AT_perror("ATL_init2()");
17:             exit(1);
18:      }
19:
20:      idata = 1;
21:      AT_strtorm(rdata, 2, "0.5V", "1.2V");
22:      if(ATL_proc(EM_ATL, "P1", INTE(&idata), DREAL(rdata, 2)) != 0){
23:          AT_perror("P2");
24:          exit(1);
25:      }
26:
27:      return(0);
28:      }
```

FIG.4


START
S121
CREATE ATL SOURCE FILE
S122
COMPILE C SOURCE SECTION
(CREATE C OBJECT CODE)
S123
COMPILE ATL SOURCE FILE
(CREATE ATL OBJECT CODE)
S124
CREATE C+ATL OBJECT FILE
RETURN
ATL SOURCE FILE
PRO SAMPLE
x.asc
C OBJECT FILE
ATL OBJECT FILE
C+ATL OBJECT FILE

Hironori MAEDA
PROGRAM DEVELOPMENT SUPPORTING..
April 18, 2005
Alan J. Kasper 202-293-7060
Q87386
5 of 12

10/531738

FIG.5

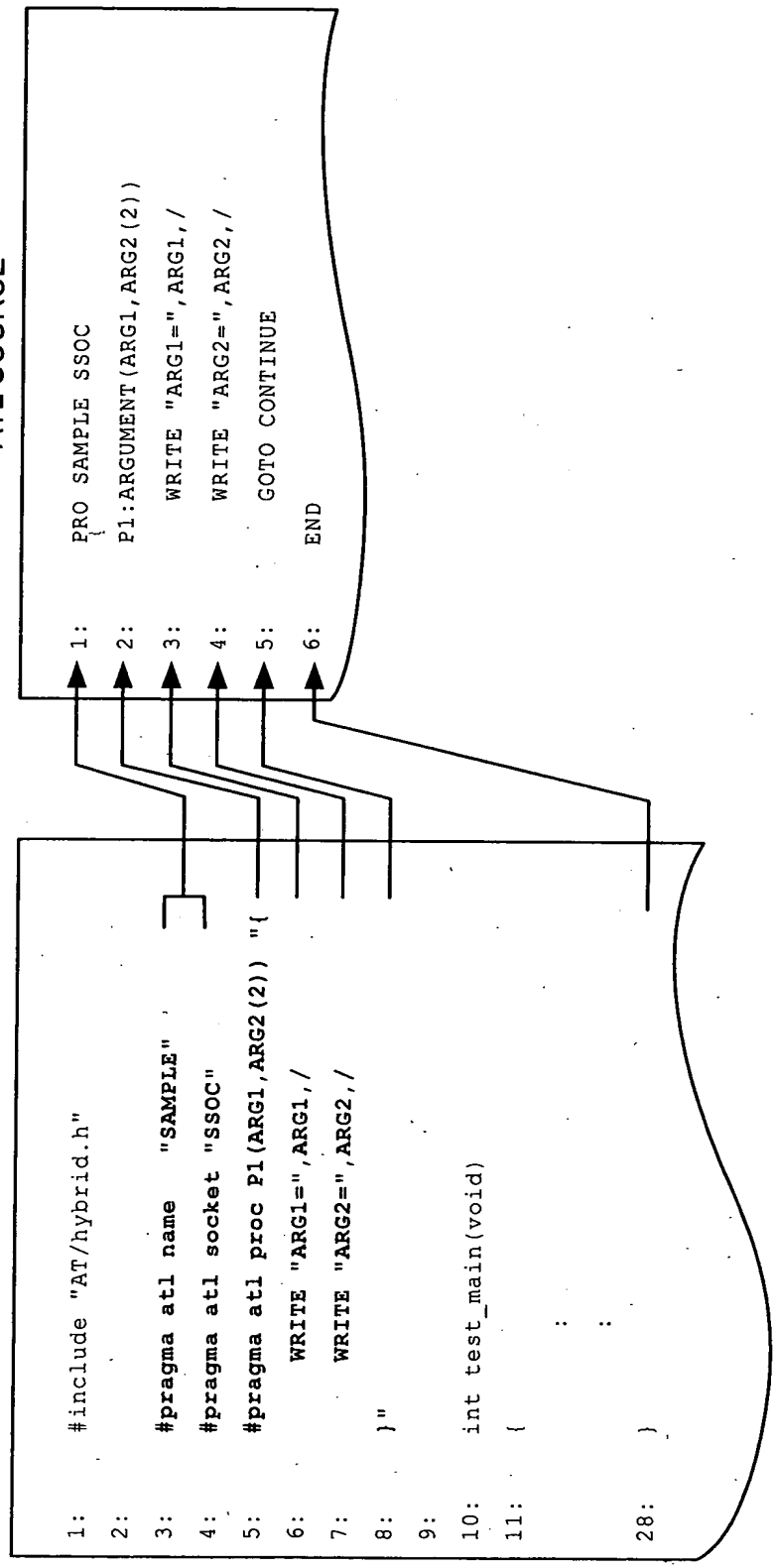


FIG.6

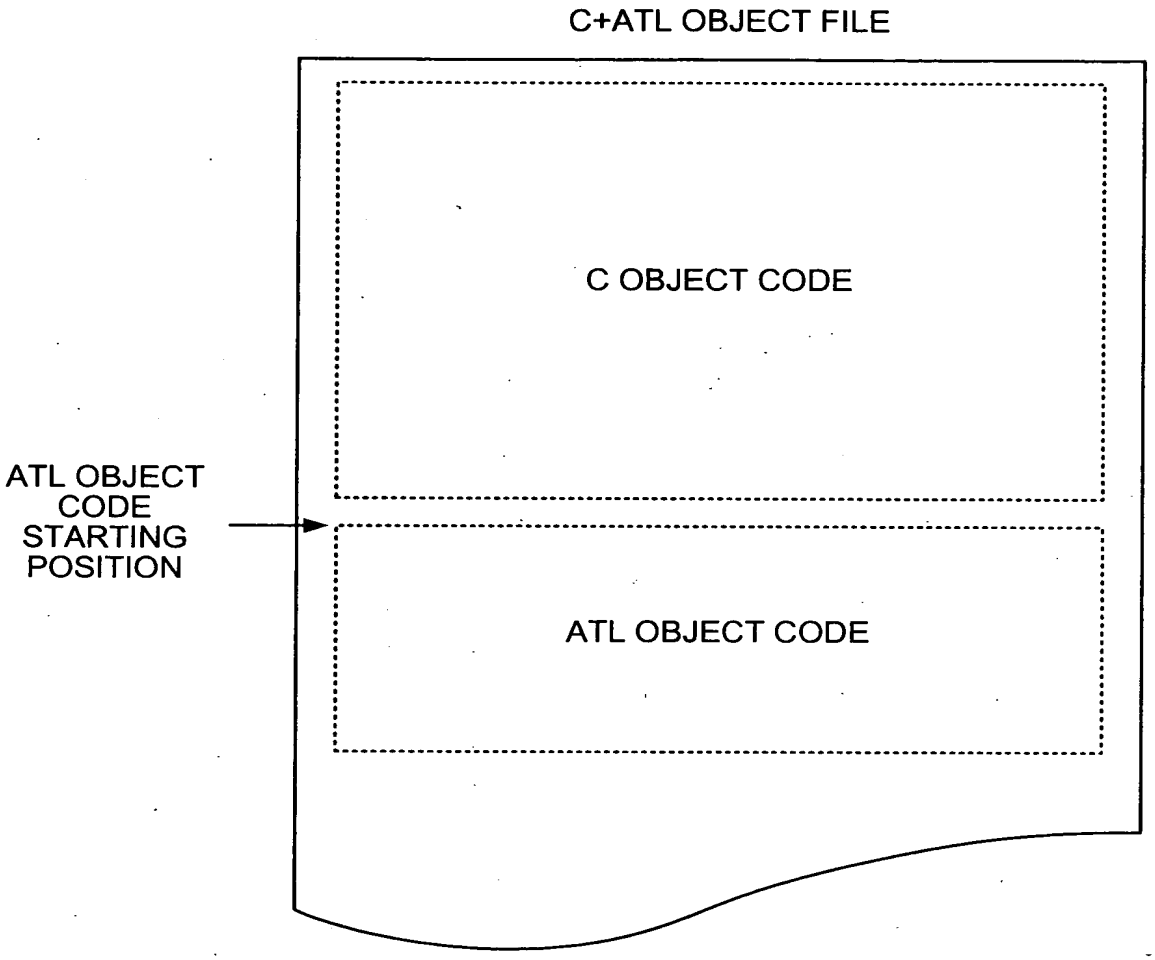

C+ATL OBJECT FILE
C OBJECT CODE
ATL OBJECT CODE STARTING POSITION
ATL OBJECT CODE

FIG.7

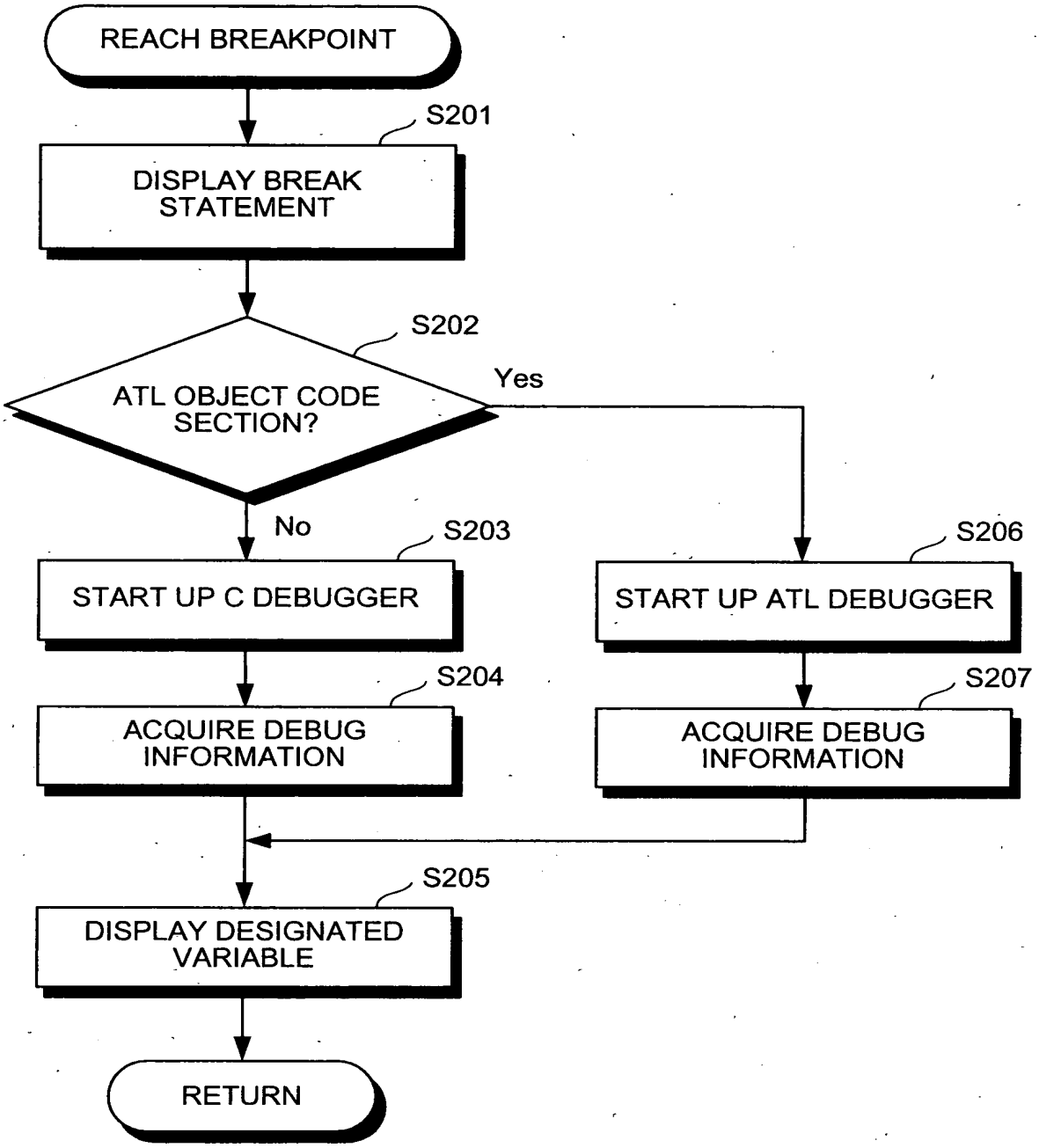
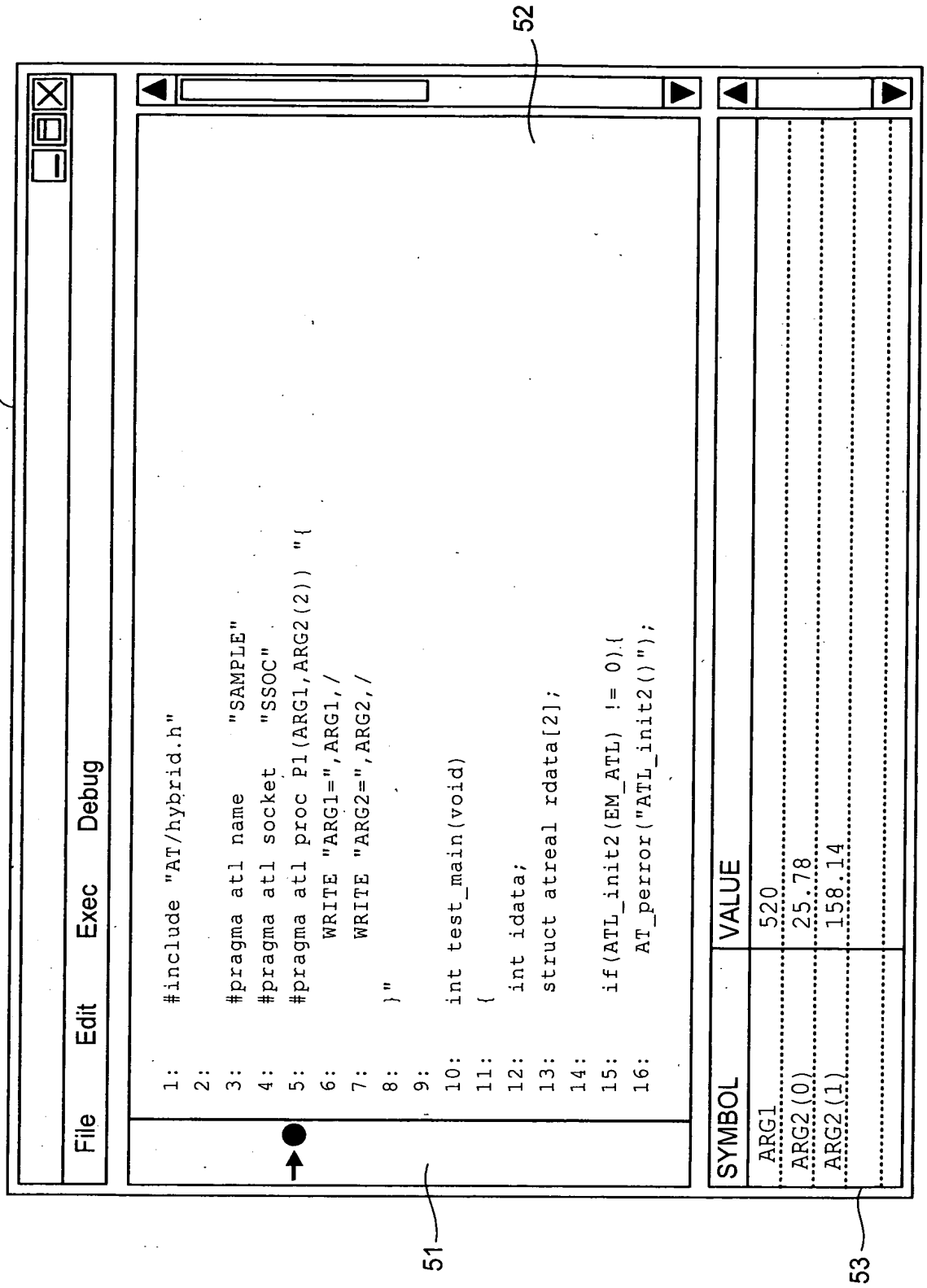

REACH BREAKPOINT
S201
DISPLAY BREAK STATEMENT
S202
ATL OBJECT CODE SECTION?
Yes
No
S203
START UP C DEBUGGER
S204
ACQUIRE DEBUG INFORMATION
S206
START UP ATL DEBUGGER
S207
ACQUIRE DEBUG INFORMATION
S205
DISPLAY DESIGNATED VARIABLE
RETURN

FIG.8


50
52
File
Edit
Exec
Debug
1: #include "AT/hybrid.h"
2:
3: #pragma atl name "SAMPLE"
4: #pragma atl socket "SSOC"
5: #pragma atl proc P1(ARG1,ARG2(2)) "{
6: WRITE "ARG1=",ARG1,/
7: WRITE "ARG2=",ARG2,/
8: }"
9:
10: int test_main(void)
11: {
12: int idata;
13: struct atreal rdata[2];
14:
15: if(ATL_init2(EM_ATL) != 0){
16: AT_perror("ATL_init2()");
SYMBOL
VALUE
ARG1
520
ARG2(0)
25.78
ARG2(1)
158.14
51
53

Hironori MAEDA
PROGRAM DEVELOPMENT SUPPORTING..
April 18, 2005
Alan J. Kasper 202-293-7060
Q87386
9 of 12

10/531738

FIG.9

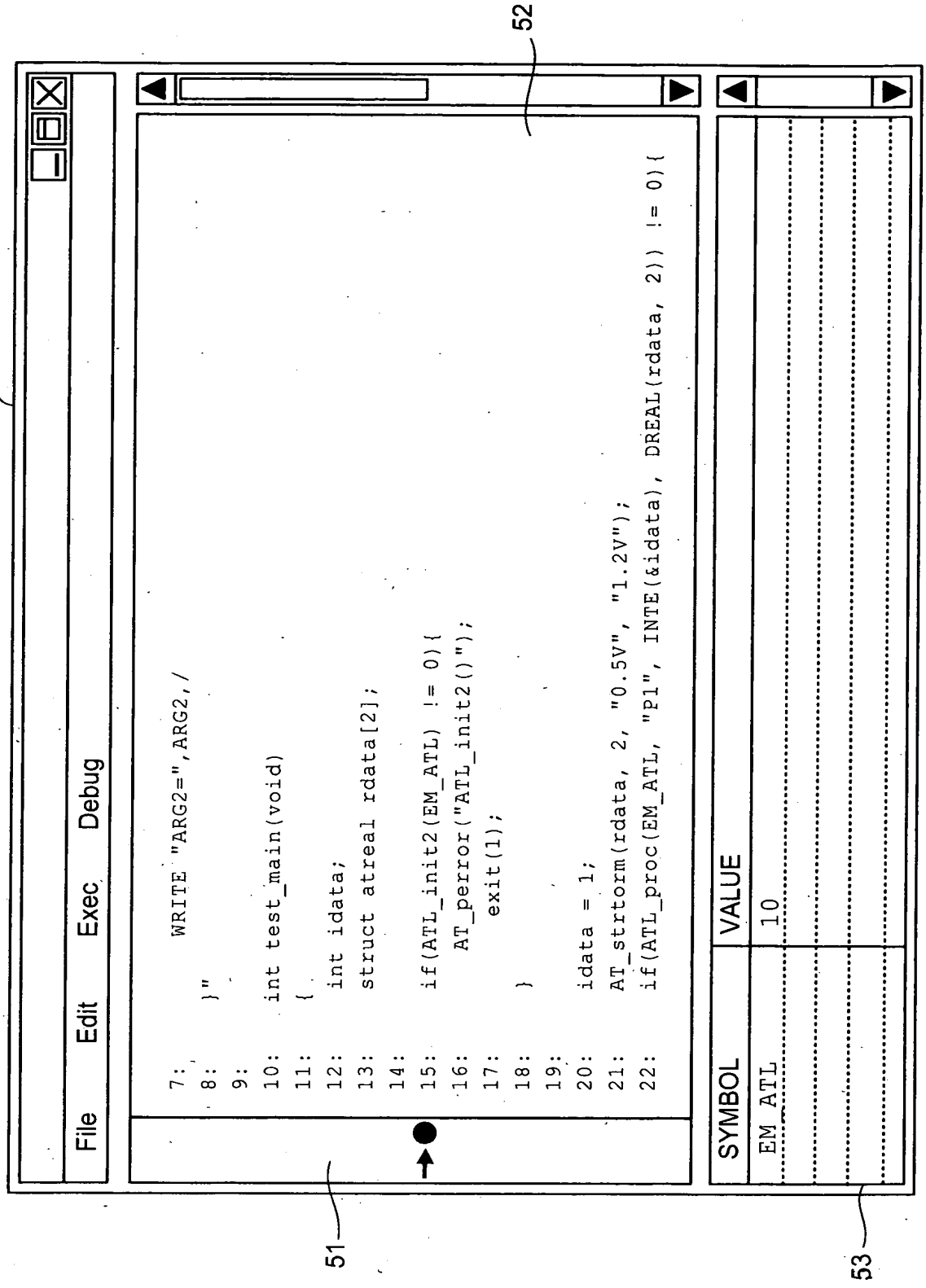


FIG.10

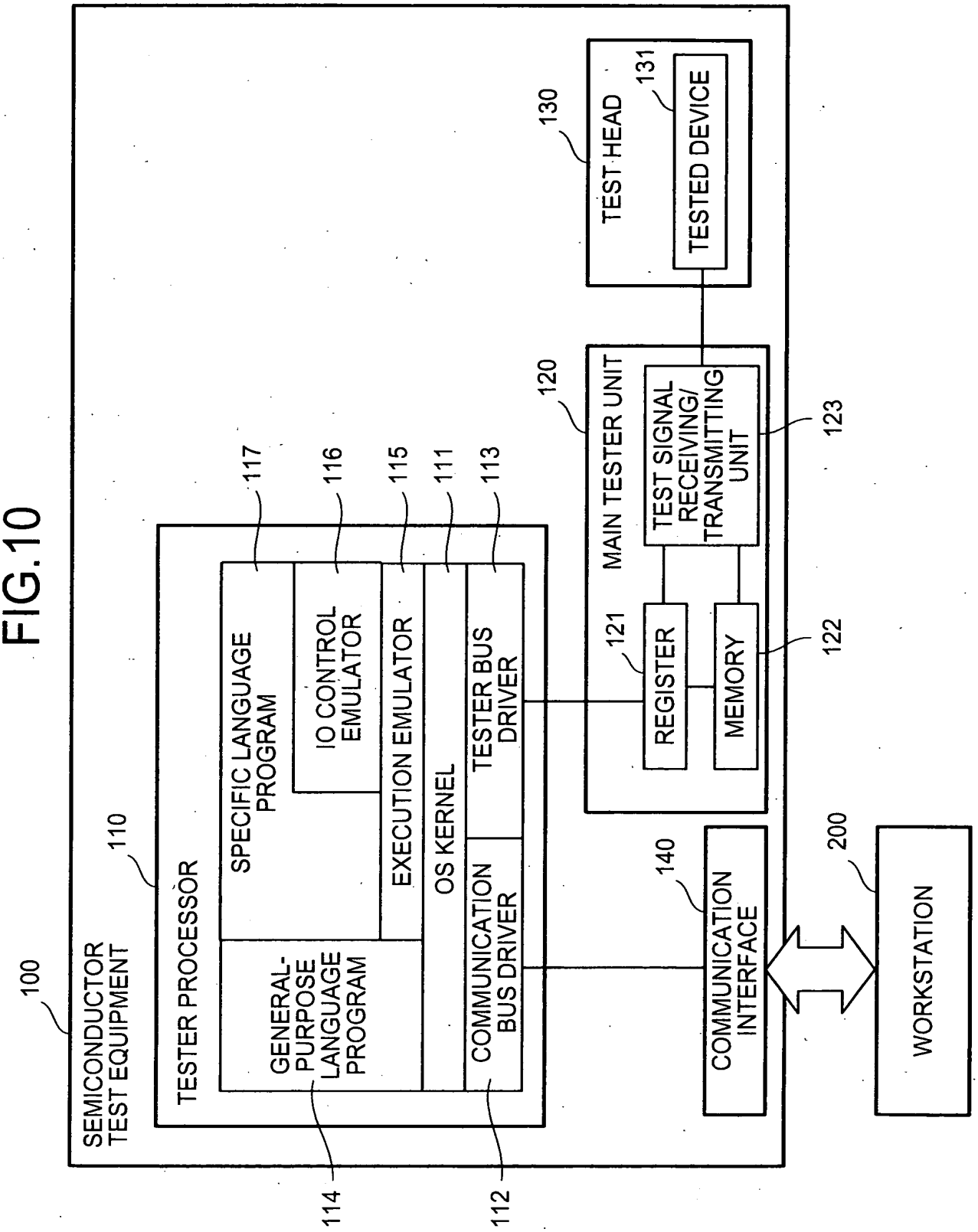

SEMICONDUCTOR TEST EQUIPMENT
100
TESTER PROCESSOR
110
SPECIFIC LANGUAGE PROGRAM
117
IO CONTROL EMULATOR
116
EXECUTION EMULATOR
115
OS KERNEL
111
TESTER BUS DRIVER
113
GENERAL-PURPOSE LANGUAGE PROGRAM
114
COMMUNICATION BUS DRIVER
112
MAIN TESTER UNIT
120
REGISTER
121
MEMORY
122
TEST SIGNAL RECEIVING/ TRANSMITTING UNIT
123
TEST HEAD
130
TESTED DEVICE
131
COMMUNICATION INTERFACE
140
WORKSTATION
200

FIG.11

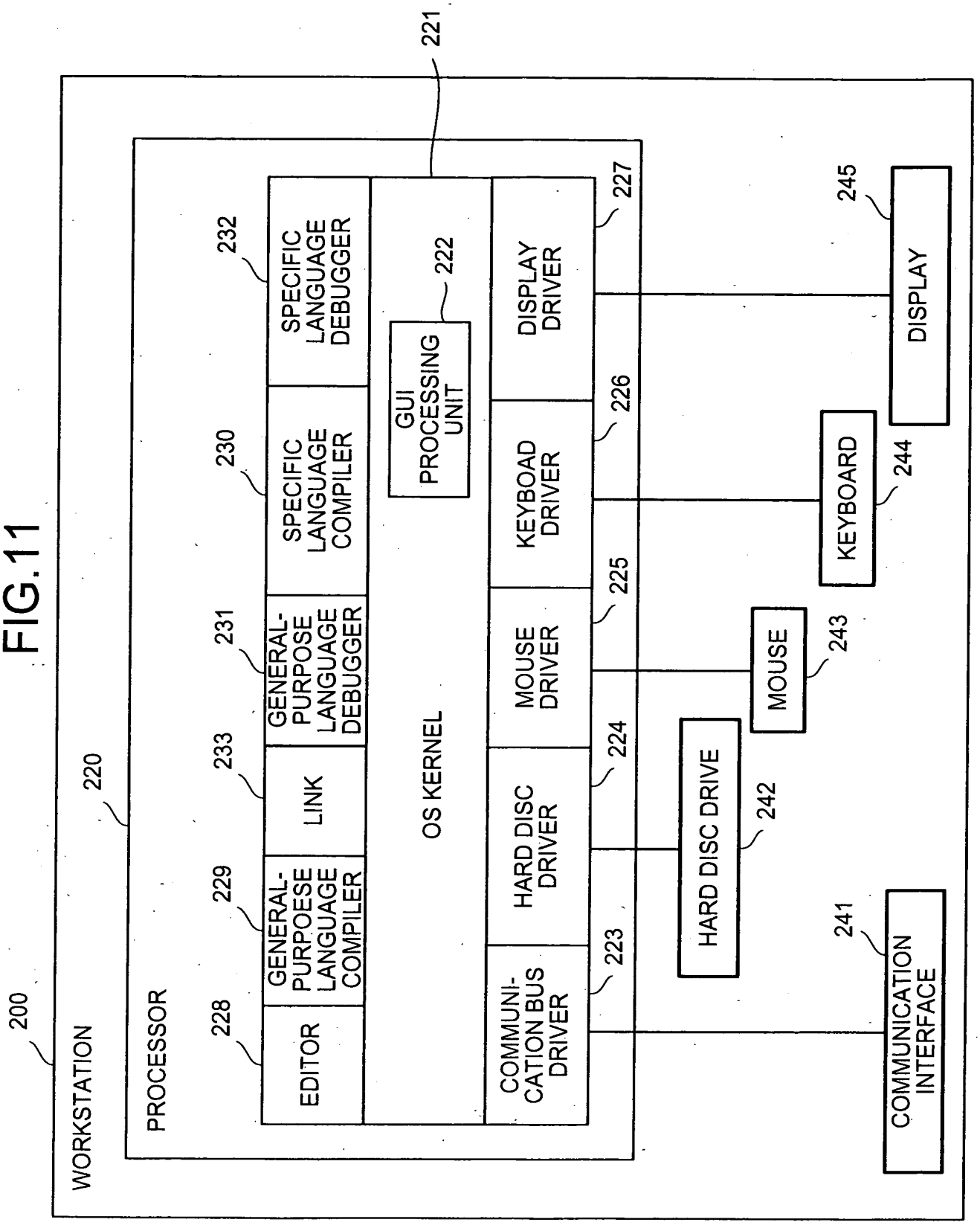
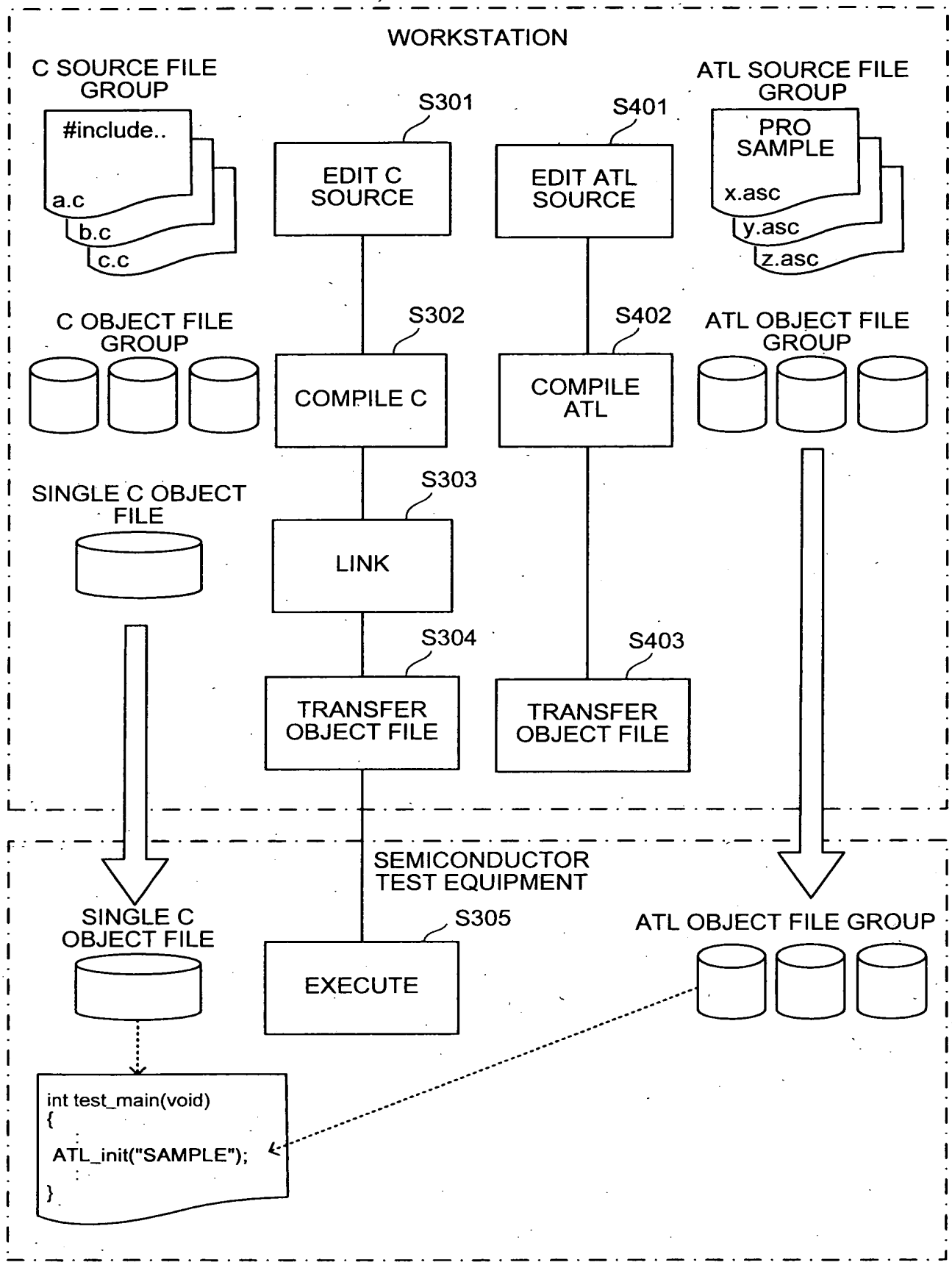

200
WORKSTATION
220
PROCESSOR
221
OS KERNEL
222
GUI PROCESSING UNIT
223
COMMUNI-CATION BUS DRIVER
224
HARD DISC DRIVER
225
MOUSE DRIVER
226
KEYBOAD DRIVER
227
DISPLAY DRIVER
228
EDITOR
229
GENERAL-PURPOESE LANGUAGE COMPILER
230
SPECIFIC LANGUAGE COMPILER
231
GENERAL-PURPOSE LANGUAGE DEBUGGER
232
SPECIFIC LANGUAGE DEBUGGER
233
LINK
241
COMMUNICATION INTERFACE
242
HARD DISC DRIVE
243
MOUSE
244
KEYBOARD
245
DISPLAY

FIG.12


WORKSTATION
C SOURCE FILE GROUP
#include..
a.c
b.c
c.c
S301
EDIT C SOURCE
S401
EDIT ATL SOURCE
ATL SOURCE FILE GROUP
PRO SAMPLE
x.asc
y.asc
z.asc
C OBJECT FILE GROUP
S302
COMPILE C
S402
COMPILE ATL
ATL OBJECT FILE GROUP
SINGLE C OBJECT FILE
S303
LINK
S304
TRANSFER OBJECT FILE
S403
TRANSFER OBJECT FILE
SEMICONDUCTOR TEST EQUIPMENT
SINGLE C OBJECT FILE
S305
EXECUTE
ATL OBJECT FILE GROUP
int test_main(void)
{
ATL_init("SAMPLE");
}